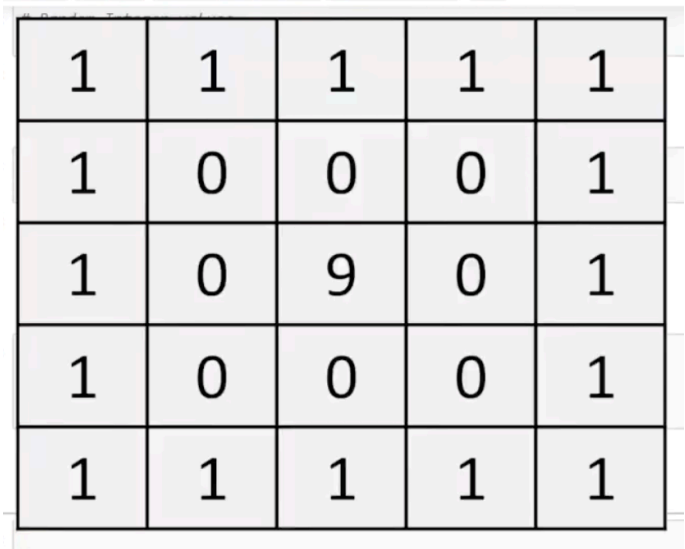
- ***-> an array we want to initialise***



1	1	1	1	1
1	0	0	0	1
1	0	9	0	1
1	0	0	0	1
1	1	1	1	1

- ***-> putting an array together without outputting all of the columns***
  - -> output = np.ones((5,5))
    - -> an entire 5x5 array of ones
    - -> we first make a 5x5 array

- ***-> z = np.zeros((3x3))***
  - -> this is the 3x3 array of zeros in the middle of it
  - -> then we fill in the middle element with a 9 -> z[1,1]=9

- ***-> then we replace the middle part of the 1's matrix with this zeros matrix***
  - -> output[1:3,1:4] = z
  - -> print(output)
  - -> this returns the matrix we wanted
  - -> you can also use ...-1] and it doesn't change

- ***-> question***

What is another way to produce the following array?

[[0. 0. 0. 0. 0. 0. 0.]
[0. 1. 1. 1. 1. 1. 0.]
[0. 1. 1. 1. 1. 1. 0.]
[0. 1. 1. 5. 1. 1. 0.]
[0. 1. 1. 1. 1. 1. 0.]
[0. 1. 1. 1. 1. 1. 0.]
[0. 0. 0. 0. 0. 0. 0.]]

output = np.ones((7, 7))

z = np.zeros((5, 5))
z[2, 2] = 5

output[1:1, -1:-1] = z

output = np.zeros((7,7)) <- This one

z = np.ones((5, 5))

```
z[2, 2] = 5

output[1:-1, 1:-1] = z

output = np.ones((7, 7))

z = np.zeros((5, 5))
z[3, 3] = 5

output[1:-1, 1:-1] = z
```